

**DEPARTAMENTO DE ELECTRÓNICA Y AUTOMÁTICA**

**FACULTAD DE INGENIERÍA – UNIVERSIDAD NACIONAL DE SAN JUAN**

Informe de Practica Nº3

CONTROLADORES DE PARÁMETROS DISCRETIZADOS

**Asignatura:** CONTROL 3

**Ingeniería Electrónica**

***Autores (Grupo Nº 4):***

*Albornoz Rubén Fernando - Registro 9827*

*Avila Juan Agustín - Registro 26076*

**1º Semestre**

**Año 2020**

# Planta Hidráulica.

Implemente un controlador PID con aproximación rectangular para la siguiente planta:

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

## Determine el período de muestreo y justifique la elección del mismo.

Primero se obtienen los polos de G(s). Para ello, se utiliza el siguiente comando de matlab:

%% punto 1.2

p=pole(G) %Se obtienen los polos de la planta

Obteniendo el siguiente resultado:

p =

-2

-1

Por lo tanto, la menor constante de tiempo será dada por el polo que está mas alejado del origen, es decir:

Por el teorema de Shannon se sabe que el tiempo máximo de muestreo no puede ser mayor a 0,5 Tmin, en este caso se elige un valor de muestreo cinco veces menor al Tmin:

## Explicite la función de transferencia discreta en forma analítica. Emplear el comando ‘c2dm’ de Matlab para comprobar el valor obtenido.

Para obtener la transformada z por convolución, se utiliza la siguiente formula:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

Al utilizarse un retenedor de orden cero, se debe multiplicar la ecuación anterior por la función de transferencia del ZOH:

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

Luego, se calculan los residuos usando la sig. fórmula:

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

El segundo residuo será:

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

El tercer residuo será:

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

*Reemplazando los valores de las ecuaciones (4) (5) y (6) en la ecuación (2), se tiene:*

*En matlab me da un valor en el denominador igual, pero el numerador 1000 veces mas chico haciendo la función de transferencia con el polo en cero, y después multiplicándolo por 1-z^-1.*

*Usando el comando c2dm me da un resultado completamente distinto. Lo comprobé haciendo la inversa de Laplace y transformándola a z en wólfram alpha, y coincidia con lo de c2dm*

La transformada z agregando el polo en el origen resulta:

|  |  |  |
| --- | --- | --- |
|  |  | (7) |

Multiplicando por 1-z^-1:

|  |  |  |
| --- | --- | --- |
|  |  | (8) |

Reemplazando T0:

|  |  |  |
| --- | --- | --- |
|  |  | (9) |

Normalizando, queda:

|  |  |  |
| --- | --- | --- |
|  |  | (10) |

Para comprobar en matlab:

[nd,dd]=c2dm(n,d,T0,'zoh')

Gz=c2d(G,T0,'zoh')

nd = 0 0.0045 0.0041

dd = 1.0000 -1.7236 0.7408

Gz = 0.004528 z + 0.004097

-----------------------------

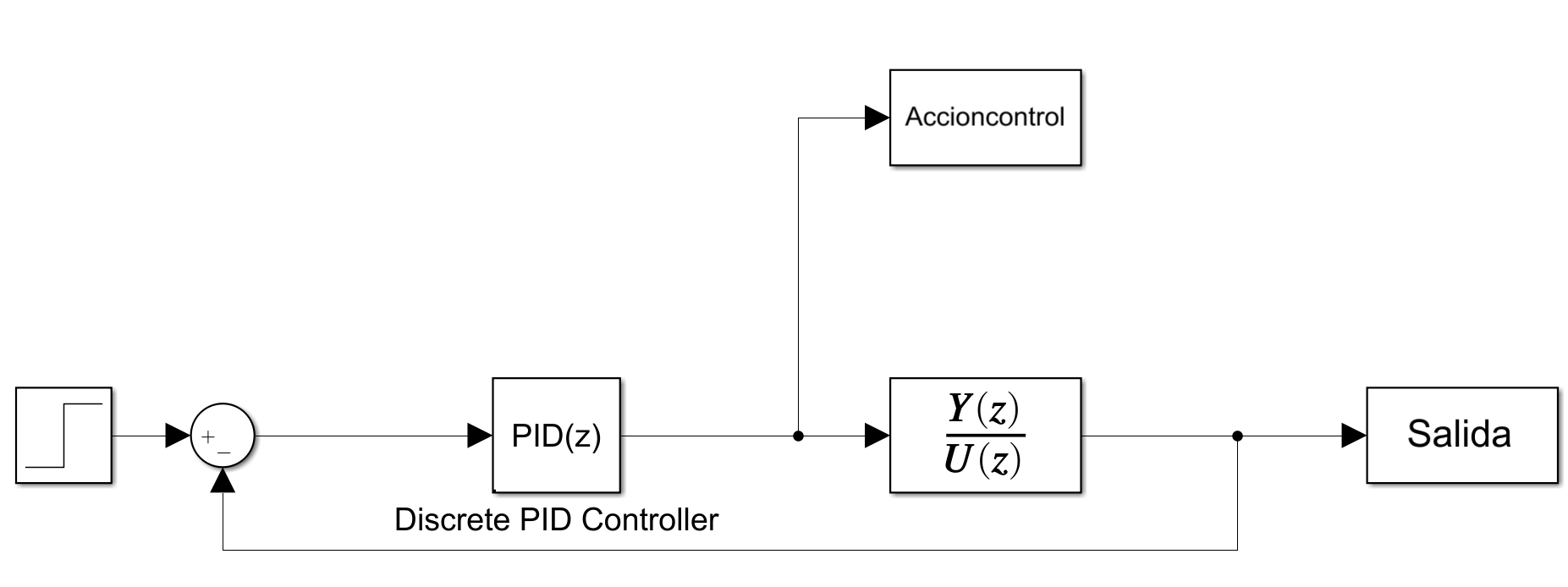
z^2 - 1.724 z + 0.7408

Sample time: 0.1 seconds

Discrete-time transfer function.

## Calcule los parámetros del controlador empleando prueba y error de la siguiente manera: emplee gráficas superpuestas de Matlab para mostrar el efecto de cambiar cada uno de los términos comenzando por el P, luego el D y finalmente el I (con al menos 5 valores de cada uno e indicando la tendencia de aumento de la constante).

Para esto, se armó la siguiente planta en simulink:



Dentro del controlador PID, se declararon kp,ki y kd como variables del entorno de trabajo, y lo mismo se realizó para la función de transferencia discreta, definiendo al numerador y denominador como las variables obtenidas del comando c2dm. Además, en todos los bloques el tiempo de muestreo se definio con la variable “T0”.

Se utilizó el siguiente código de matlab:

%% punto 1.3

ki=0;kd=0; %define kd y ki=0 para variar solo kp

mkp=2; %los m son los "pasos" que dara en cada prueba

nkp=2; %para fijar una constante, se elige m\*n

mkd=2; %En este caso, kp=mkp\*nkp=2\*2=4, kd=2\*1=2

nkd=1; %se usaron estas variables para facilitar

mki=1; %el uso de distintos rangos

figure();

Legend=cell(5,2);% Arreglo de celdas para nombrar señales en la grafica

for i=1:5 %hace 5 iteraciones, desde 1 hasta 5

kp=mkp\*i; %kp varia desde mkp\*1 hasta mkp\*5

sim('punto1.slx'); %simula la planta en simulink

subplot(2,1,1); %grafica la salida en la parte superior

plot(Salida);

Legend{i,1}="Salida con Kp="+kp;

hold on;

subplot(2,1,2);

plot(Accioncontrol);%Grafica la accion de control

hold on;

Legend{i,2}="Accion con Kp="+kp;

end

subplot(2,1,1); %fuera del bucle, se vuelve a la grafica superior

title('Salida de la planta variando el Kp');

legend(Legend{:,1});grid on; %agrega todos los nombres de las señales

subplot(2,1,2);

title('Accion del controlador variando el Kp');

legend(Legend{:,2});grid on;

%% para kd

kp=mkp\*nkp; %Fija el Kp en un valor dado por m\*n

figure();

Legend=cell(6,2);

for i=0:5 %En este caso se comienza el bucle desde cero para

kd=mkd\*i; %graficar la respuesta sin la accion derivativa

sim('punto1.slx');

subplot(2,1,1);

plot(Salida);

Legend{i+1,1}="Salida con Kd="+kd;

hold on;

subplot(2,1,2);

plot(Accioncontrol);

hold on;

Legend{i+1,2}="Accion con Kd="+kd;

end

subplot(2,1,1);

title('Salida de la planta variando el Kd');

legend(Legend{:,1});grid on;

subplot(2,1,2);

title('Accion del controlador variando el Kd');

legend(Legend{:,2});grid on;

%% para ki (procedimiento igual a kd)

kd=mkd\*nkd;

figure();

Legend=cell(6,2);

for i=0:5

ki=mki\*i;

sim('punto1.slx');

subplot(2,1,1);

plot(Salida);

Legend{i+1,1}="Salida con Ki="+ki;

hold on;

subplot(2,1,2);

plot(Accioncontrol);

hold on;

Legend{i+1,2}="Accion con Ki="+ki;

end

subplot(2,1,1);

title('Salida de la planta variando el Ki');

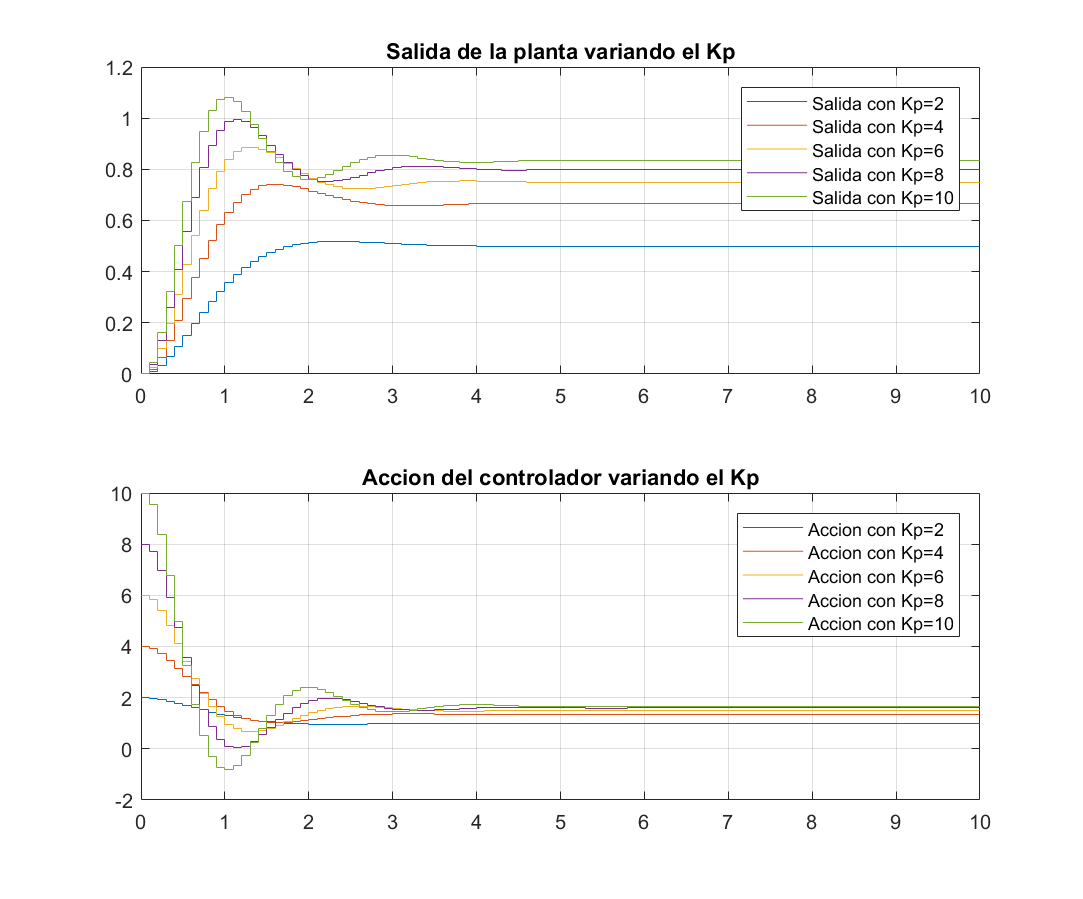
legend(Legend{:,1});grid on;

subplot(2,1,2);

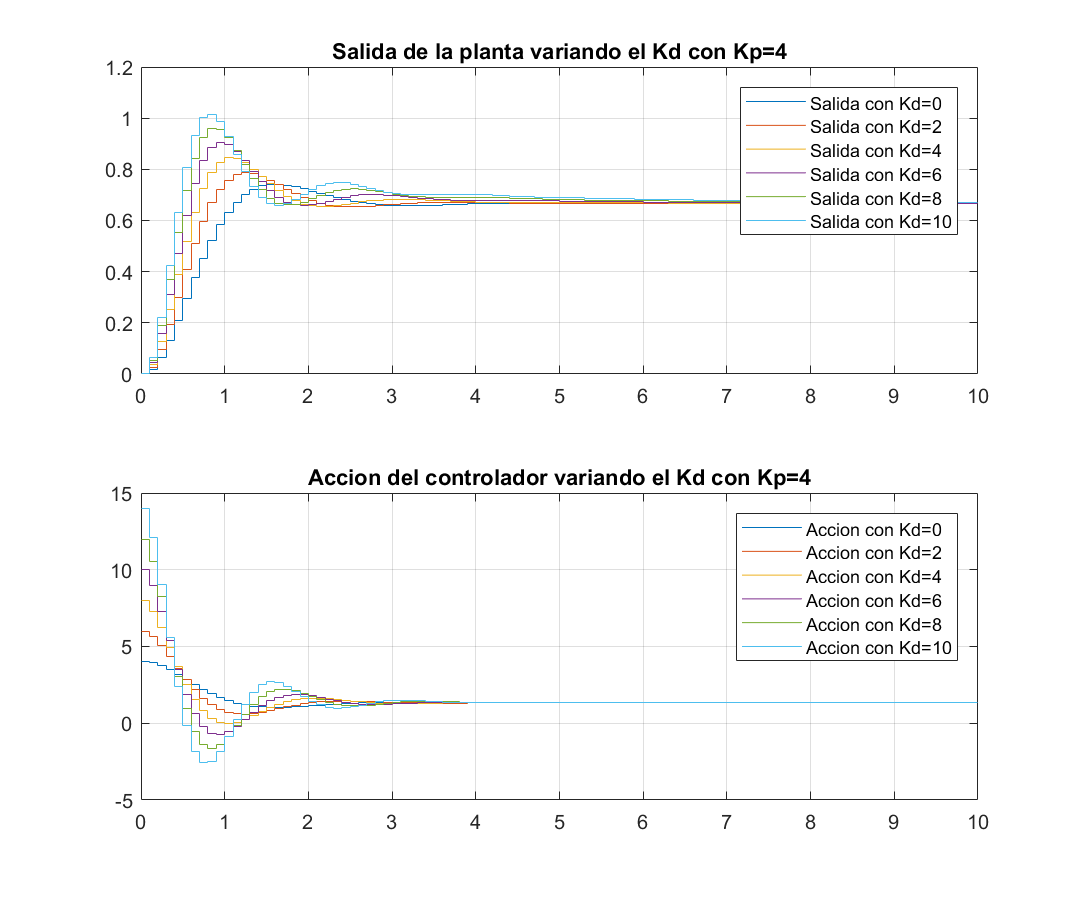
title('Accion del controlador variando el Ki');

legend(Legend{:,2});grid on;

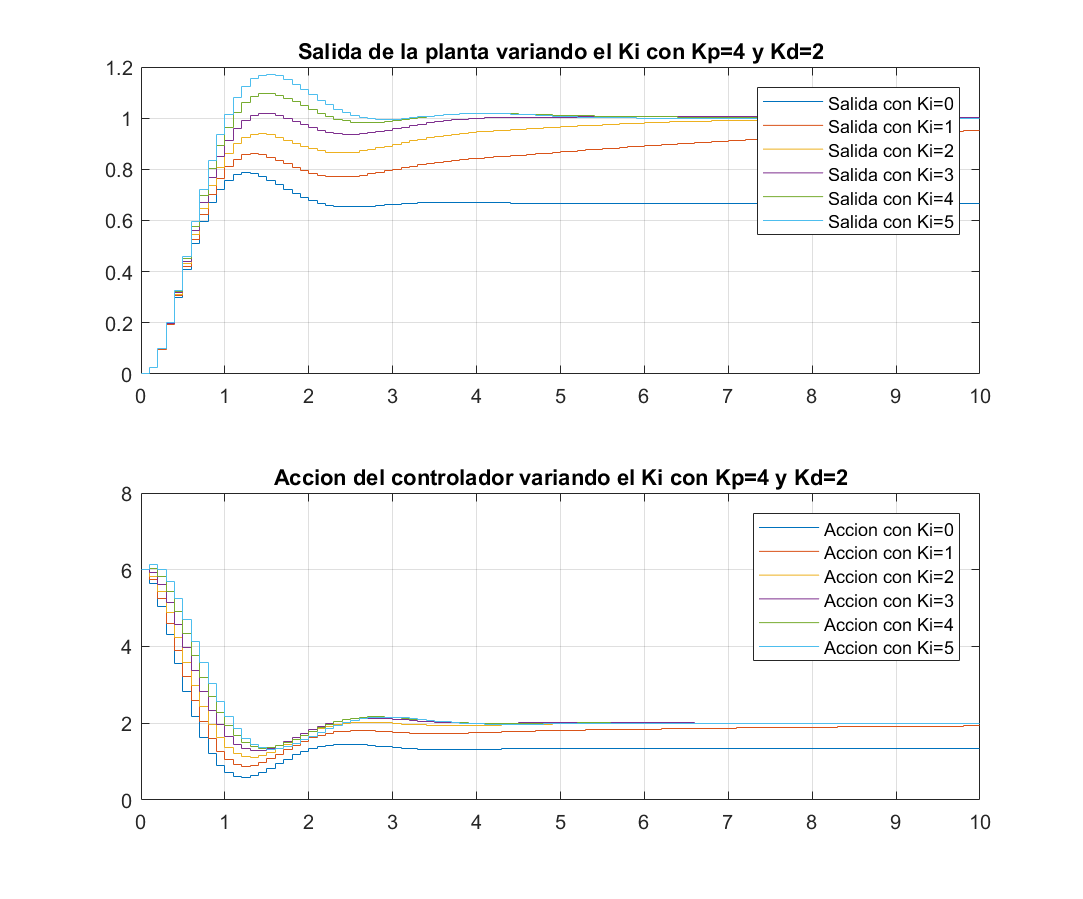
Las distintas salidas se reproducen a continuación:



En este caso, se observa que a mayor Kp, la respuesta del sistema a una entrada escalón es cada vez más rápida y el error en estado estacionario es menor, pero también a medida que se aumenta el Kp, aumenta el sobreimpulso y también aumenta la acción de control. En este caso, se eligió un Kp=4, ya que la respuesta no es tan lenta, el sobreimpulso es bastante pequeño, la acción de control no toma un valor tan grande y

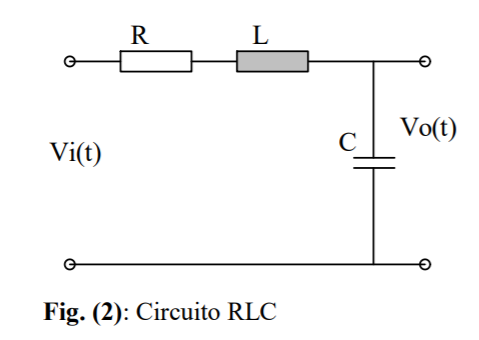


(descripción de la acción derivativa)



(Descripcion de la acion integral)

# 2 PLANTA ELÉCTRICA DEL TP2. PID discreto.



Donde: R =10Ω L=10mHy C=10μF Vi(t)=10V

## 2.1 Determine el período de muestreo y justifique la elección del mismo.

Se sabe del TP2 que la función de transferencia de la planta eléctrica es la siguiente:

Los polos están ubicados en 1.0e+03 \*

-0.5000 + 3.1225i

-0.5000 - 3.1225i

Que es un par de polos complejos conjugados con parte real igual a -500. Por lo tanto su periodo es 1/500=2ms. Para cumplir con el teorema de Shannon, el tiempo de muestreo debe ser al menos la mitad del T de la planta, es decir, 1ms. En este caso, se selecciona una frecuencia de muestreo 5 veces mayor que la frecuencia de Nyquist, es decir:

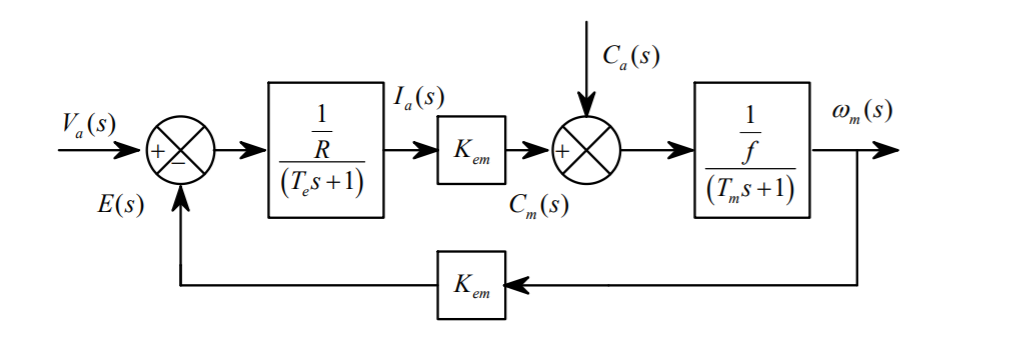
|  |  |  |
| --- | --- | --- |
|  |  | (11) |

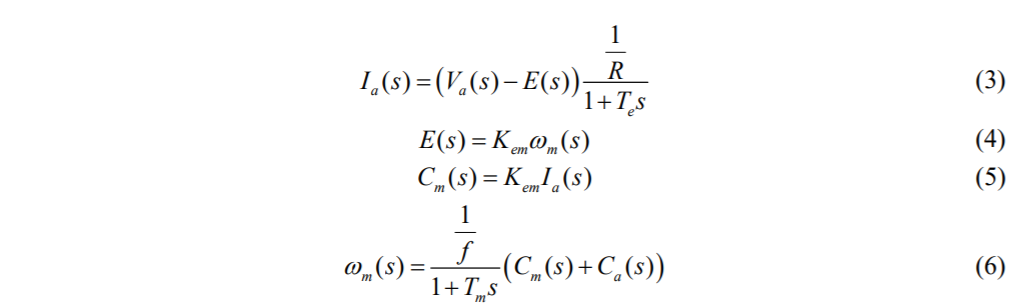
## 2.2 Calcule los parámetros del controlador mediante Ziegler-Nichols de lazo abierto. Muestre los resultados obtenidos en gráficas de salida para entrada escalón y luego para las acciones de control de manera que se puedan ver en forma simultánea, pero no solapada. (emplear subplot, y ubicar una debajo de la otra, que se pueda ver).

## 2.3 Recalcule por prueba y error hasta obtener un valor adecuado

# MOTOR DE DC DEL TP2. PID convencional vs PID modificado.

Modelo de un motor de corriente continua expresado en las Ecs. (3), (4), (5) y (6) y esquematizado en el diagrama de bloques presentado en la Fig. (3)





La alimentación del motor es una tensión continua de 100 V, la carga mecánica es un par antagónico de 1 Nt aplicada a los 0.15 s del arranque.

## Determine el período de muestreo y justifique la elección del mismo.

Se cargan las constantes en matlab y se genera la función de transferencia:

%% punto 3

R=8;L=.08;Te=L/R;Kem=0.67;

J=2.22\*10^-3;f=1.86\*10^-3;Tm=J/f;

Va=100;Tcarga=0.15;

G1=tf(1/R,[Te 1]);

G2=tf(1/f,[Tm 1]);

H=feedback(G1\*G2\*Kem,Kem)

[n,d]=tfdata(H,'v');

p=pole(H)

La función de transferencia del motor para la entrada de referencia es la siguiente:

|  |  |  |
| --- | --- | --- |
|  |  | (12) |

Y sus polos están ubicados en:

p=

-50.4189 + 8.3251i

-50.4189 - 8.3251i

Por lo tanto, su constante temporal es 1/50=20ms. Se selecciona un periodo de muestreo 5 veces menor, por lo tanto T0=4ms.

## Emplee Ziegler-Nichols de lazo cerrado para ajustar el PID convencional y Takahashi para el modificado. Muestre los resultados. Ajuste por prueba y error a la mejor respuesta posible para cada uno.

## Presente una comparación entre ambos.

# INDICE DE DESEMPEÑO.

Para alguno de los controladores PID ajustados, programa un algoritmo recursivo para encontrar los valores de las constantes tales que minimicen algún índice propuesto por ustedes 𝐽, el cual debe depender al menos del error cuadrático medio. Tomen una variación de 20% para cada uno de los valores ajustados y presenten gráficas de:

## Código .m del algoritmo programado.

## Variación del índice para todos los casos.

## Comparación de la respuesta con mejor índice vs la respuesta obtenida en el ejercicio elegido